# A collision-free graph coloring MAC protocol for underwater sensor networks

Alfouzan, Faisal Abdulaziz; Shahrabi-Farahani, Alireza; Ghoreyshi, Seyed Mohammad; Boutaleb, Tuleen

*Published in:*
IEEE Access

*DOI:*
10.1109/ACCESS.2019.2906555

*Publication date:*
2019

*Document Version*
Publisher's PDF, also known as Version of record

IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received February 1, 2019, accepted March 9, 2019, date of publication March 27, 2019, date of current version April 8, 2019.

*Digital Object Identifier 10.1109/ACCESS.2019.2906555*

# A Collision-Free Graph Coloring MAC Protocol for Underwater Sensor Networks

**FAISAL ABDULAZIZ ALFOUZAN, (Member, IEEE), ALIREZA SHAHRABI, (Member, IEEE), SEYED MOHAMMAD GHOREYSHI, (Member, IEEE), AND TULEEN BOUTALEB, (Member, IEEE)**
School of Computing, Engineering and Built Environment, Glasgow Caledonian University, Glasgow G4 0BA, U.K.
Corresponding author: Faisal Abdulaziz Alfouzan (faisal.alfouzan@gcu.ac.uk)

**ABSTRACT** Employing contention-based medium access control (MAC) protocols in underwater sensor networks (UWSNs) is typically costly. This is due to the unique characteristics of underwater acoustic channels, such as long propagation delay, limited bandwidth, and high bit error rate. As a consequence, the contention-based (handshaking and random access-based) MAC protocols do not perform as efficiently as expected. The collision-free approach is therefore considered to achieve high performance by avoiding the collisions at the MAC layer in order to improve energy efficiency, throughput, and fairness. In this paper, we propose, inspired by the graph coloring techniques, a novel energy-conserving and collision-free reservation-based MAC protocol, called GC-MAC, for UWSNs. GC-MAC employs time-division multiple access (TDMA)-like approach by assigning separate time-slots, colors, to every individual sensor node in every two-hop neighborhood. Sensors with the same colors can thus transmit at the same time with no chance of collision. GC-MAC is also able to address the near–far effect, spatial–temporal uncertainty, and hidden/exposed node problems, without requiring code-division multiple access (CDMA) or power adjustment for collision avoidance. The network coverage and connectivity is then discussed to show the effectiveness of using cubes to cover a 3D underwater environment. Our extensive performance study shows that GC-MAC performs well by avoiding collisions to achieve better throughput and energy-efficiency performance compared with those of contention-based protocols. There is also a significant improvement in terms of packet delivery ratio and fairness among the nodes under different operational conditions.

**INDEX TERMS** Underwater sensor networks (UWSNs), medium access control (MAC), collision-free MAC protocols, graph coloring technique, distributed clustering approach.

## I. INTRODUCTION

Underwater sensor networks (UWSNs) have attracted a considerable attention over the last decade. This is due to the wide range of applications that are used, such as oceanographic data collection, environmental monitoring, early warning systems, assisted navigation, tactical surveillance, and resource discovery [1]–[4]. Due to the harsh environment of the oceans which poses some inherent characteristics of underwater acoustic channels such as high latency, high bit error rate, and low available bandwidth [5], [6], the design of MAC (Medium Access Control) protocols for UWSNs faces many challenges.

The MAC protocols can typically be classified into two categories: contention-based and contention-free protocols [7]. The first category, contention-based protocols, includes handshaking and random access-based MAC protocols. In handshaking protocols, sender and receiver nodes reserve the channel through small control packets before sending data packets. While in a random access-based protocol, the transmitter sends data packets with no prior coordination. A packet is successfully transmitted to a receiver as long as no other simultaneous data transmission occurs. Thus, collision avoidance is entirely a probabilistic approach [8], [9].

However, one critical issue, which forms the focus of this study, is how to improve the performance of MAC in UWSNs. In the literature, extensive studies have recently been reported to explore underwater MAC protocols. In this context, some random access-based MAC protocols have been specifically designed for UWSNs to achieve better performance, such as UWAN-MAC [10] and Aloha with Advance Notification (ALOHA-AN) [11]. Both of these

The associate editor coordinating the review of this manuscript and approving it for publication was Kaigui Bian.

leverage short control packets to reduce collisions. However, these types of solutions do not perform efficiently in multi-hop UWSNs because they cannot detect the hidden node problem [8], [12].

Since the random access-based MAC protocols are not effectively suitable for a multi-hop in UWSNs, several handshaking MAC protocols, using a Request-To-Send/ Clear-To-Send (RTS/CTS) mechanism, have been particularly designed for multi hops to achieve better performance. Floor Acquisition Multiple Access (FAMA) [13], is one of these MAC protocols, which extends the transmission delays of RTS/CTS control packets to allow Multiple Access with Collision Avoidance (MACA) [14] dealing with high latency such that in underwater acoustic communication. However, it is an energy-consuming by transmitting long control packets. Slotted Floor Acquisition Multiple Access (S-FAMA) [15] is used to improve the energy efficiency by introducing a time-slot technique to FAMA; hence, the control packets (RTS/CTS) are very short to still be able to prevent collisions in UWSNs. MACA with Multiple Neighbors (MACA-MN) [16] is another MACA-based protocol which generates multiple parallel transmissions with a single handshake to improve the channel utilization.

UWSNs are expected to achieve a better performance with the above-mentioned MAC protocols. However, some recent observations regarding factors such as long propagation delay and a high bit error rate, mean that the contention-based MAC protocols are very costly, and hence both random access-based and handshaking MAC protocols are not as effective as their achieved in terrestrial networks [5], [17]–[19].

In the second category, contention-free protocols, communication channels are divided into frequency, time, or code domains, as appeared in approaches such as Frequency Division Multiple Access (FDMA), Time Division Multiple Access (TDMA), and Code Division Multiple Access (CDMA) [20]. FDMA splits the frequency band into a number of sub-bands. Because of the narrow bandwidth in underwater channel that can only be used, FDMA achieves inefficient performance (i.e., low throughput), which is not proper for UWSNs. TDMA allows the same frequency channel to be used by dividing the signal into different time-slots. To maintain reliable transmission schedules, access to the media should be distributively coordinated. TDMA also allows sensors, located out of each others' transmission ranges, to transmit data packets simultaneously without collision. CDMA is very robust to frequency selective fading caused by multiple paths. This in turn reduces packet retransmissions and increases channel utilization. CDMA is, however, not suitable for UWSNs due to its difficulties to address the near-far problem and also to assign pseudo-random codes to large numbers of sensors [21]–[23].

Due to contentions-based MAC protocols being expensive to implement in UWSNs, a collision-free MAC is considered to significantly achieve a better performance. Besides, the long propagation delays in centralized MAC protocols typically take a long period of time to gather the global topology and transmission requests from all the sensors and then to inform them of the schedule, therefore a distributed solution is preferred. To make our proposed MAC protocol appropriate for most existing acoustic modems, dynamic power adjustment and CDMA are not required to achieve these goals. In the dynamic power adjustment technique, acoustic modems usually support a limited number of predefined power levels which may be unsuitable with the value selected by the MAC protocol. For CDMA, the available bandwidth in real acoustic modems is already extremely narrow; hence, the spreading factor is significantly extended, leading to transmission delays and degrading the ability of the network [24].

In this paper, we propose a new collision-free Graph Coloring MAC protocol (GC-MAC) which achieves better throughput, energy efficiency, and fairness than that in contention-based MAC protocols. First, GC-MAC is inspired by graph coloring techniques to achieve as many concurrent conflict-free transmissions as possible in any two-hop neighbourhood. Second, by scheduling the transmissions and receptions of data packets at both the sender and receiver sides, nodes can properly achieve the objectives of high throughput, energy efficiency, efficient channel utilisation, and fairness. This is achieved by using a distributed clustering approach for up to two-hop neighbouring nodes and then to address the hidden and expose node problems by removing the possible color conflict in two-hop neighbouring graph. Finally, using a TDMA-like approach, GC-MAC is able to assign time-slots, colors, to every individual node in any two-hop neighbourhood in a distributed manner. Sensor nodes with the same colors can thus transmit concurrently without collision to support spatial reuse (simultaneous transmissions in different neighborhoods). Sensor nodes are awake in some time-slots to transmit or receive data packets and asleep over the remaining time-slots.

This paper is an extended version of our previous work [19]. The present paper extends the conference version with formulating the MAC challenges, discussing the problem complexity under very specific circumstances, and proposing a new conflict detection mechanism. The graph coloring scheme is also discussed with more details, examples, and figures. The network coverage and connectivity in three-dimensional (3D) area are discussed and analyzed in detail. We finally propose additional results related to the network parameters impacts such as evaluating different traffic rates and network scalability. The reminder of this paper is organized as follows: Section II presents the related work. Section III explains the underwater challenges when designing MAC protocols. Section IV describes GC-MAC protocol in detail. Section V presents two specific scenarios of collisions followed by a description of how these scenarios are resolved. Section VI discusses and analyses the network coverage and connectivity. Section VII evaluates the performance of our proposed protocol against other MAC protocols through simulations. Finally, Section VIII draws the conclusions.

## II. RELATED WORK

Energy consumption is a significant issue in UWSNs since increased energy consumption generally shortens the battery life of each sensor (i.e., transmitting a data packet consumes more energy in UWSNs). On the other hand, replacing or recharging of exhausted batteries of sensors is difficult and costly if not impossible. Due to these issues of energy consumption in UWSNs, most of the MAC protocols therefore try to avoid packets collisions and retransmissions.

MAC protocols can generally be divided into two categories: contention-based and contention-free. The first category, contention-based, is further classified into two classes: random access-based and handshaking MAC protocols [8], [25], [26]. The MAC protocols in the first class, random access-based, are generally a modified version of ALOHA protocol. A small control message is utilized as a transmitting notification to neighboring sensors [27]. Upon receiving the transmitted notification of other sensors, the receiver backs off its own transmission randomly based on the information received from its neighboring notifications [11], [28]. Some random access-based MAC protocols, such as UWAN-MAC [10] and Aloha with advance notification (ALOHA-AN) [11], are particularly designed for UWSNs to achieve better performance. This class with transmission notification techniques, however, wastes energy and channel bandwidth because of the packets collision caused by the hidden node problem. Some MAC protocols in the second class, handshaking, have been proposed to enhance the network performance and to also provide valid solutions by addressing the hidden node problem using RTS/CTS, such as S-FAMA [15], DACAP [29], DOTS [30], and R-MAC [22]. These protocols in this class exploit virtual carrier sense to avoid collisions and to save energy. Furthermore, they utilize small control messages, which reduce their chance of collision in comparison to regular data packets [31]. The network throughput, however, is usually low due to the high delay in the handshaking process. Due to the inherent characteristics of acoustic modems, the contentions by using RTS/CTS control messages become costly. Therefore, the contention-based MAC protocols (random access-based and handshaking classes) are not as efficiently as their achieved in terrestrial networks [17]–[19], [32].

Due to the contention-based MAC protocols are costly in UWSNs, a collision-free MAC protocol promises to avoid collisions by using scheduling-based or clustering approaches. In the scheduling-based approach, some MAC protocols guarantee collision-free transmissions [24], [33]–[38]. Among them, an Efficient Depth-based MAC protocol (ED-MAC) [33], [34] utilizes a duty cycle mechanism by allowing every sensor in the network to assign a time-slot based on its priority in a distributed manner. It is, therefore, considered as a collision-free protocol by employing the concept of sub-slots in every slot to avoid collisions between two hidden nodes that are neighbors of another deeper node. As a consequence, ED-MAC highly enhances the network performance while the number of slots is doubled in each round to avoid any chance of collision that may occur between sensors located out of 1-hop neighboring nodes and the sensors within a 1-hop neighborhood. Consequently, the channel utilization is low although the proposed MAC protocol is collision-free.

ST-MAC [36] is also a collision-free MAC protocol. It takes the advantages of using the global topology information by creating a conflict graph. Through this mechanism, ST-MAC is able to schedule all the sensor nodes with this conflict graph to enhance the network performance. However, it uses a centralized scheduling algorithm (i.e., using the global network's topology information) which is costly to collect in UWSNs because of high latency and low transmission rates. Moreover, ST-MAC always assigns time slots in a batch to be linked with the highest traffic rate to maximize the network throughput. However, ST-MAC impairs system fairness and also starves some sensor nodes. ISTLS and its variants [35] distinctly provide collision-free scheduling, but face similar problems.

STUMP [24] is a typical TDMA-based collision-free MAC protocol. In this approach, the scheduling of every node is fixed for the whole network life. This strategy considerably reduces the channel utilization if the nodes' traffic loads are significantly heterogeneous. A similar approach called UW-FLASHR [37] uses a time-based scheme which allows every sensor node to access the medium channel with a distinct size. UW-FLASHR does not require a clock synchronization or accurate propagation delays. It also operates over a period of time (i.e., cycles), while every cycle is split into established and experimental portions. Each sensor requests a time-slot to send randomly a data packet in the experimental portion. Due to a random assigning time-slot (i.e., randomly selecting a transmission time), UW-FLASH gradually generates a loose scheduling transmission that allows the time gaps between transmissions.

Some other MAC protocols utilize clustering approach to avoid collisions [39]–[41]. In [39], a multi-cluster protocol, which is an access scheme based on clustering, provides effective scalability by allowing spatial reuse of the channel resources. In particular, the neighboring nodes are grouped into clusters using TDMA for clustering and CDMA for inter-cluster communications. Nodes belonging to multiple clusters are able to use CDMA, as inter-cluster communications, to provide multi-user receiver systems. Therefore, nodes can concurrently receive data packets with different CDMA codes from multiple clusters. However, as mentioned in [24], CDMA may reduce the system performance in UWSNs because of low bandwidth characterized and packet transmission delays in UWSNs, which are considerably extended by the spreading factor.

Compared with the aforementioned collision-free in both scheduling-based and cluster-based MAC protocols, GC-MAC is able to construct collision-free scheduling in a distributed manner without requiring nodes' traffic loads in

advance nor any global topology information. It is also able to efficiently schedule inner and outer sensor nodes across the network. Furthermore, our proposed MAC protocol, GC-MAC, does not require CDMA modulation, and is still able to provide collision-free transmissions during the scheduling phase.

## III. UNDERWATER CHALLENGES FOR DESIGNING MAC PROTOCOLS

When designing a resource-sharing mechanism, it is fundamental to consider the inherent characteristics of the channel, such as high latency, low available bandwidth, and frequency-dependent attenuation. Those constraints significantly affect the MAC protocol design because of the challenges described as follows.

### A. SPATIAL-TEMPORAL UNCERTAINTY PROBLEM

To solve the collision issue in terrestrial wireless networks, it is necessary simply to restrict the interfering sensor nodes from transmitting simultaneously, since the propagation delay is negligible in this case. In UWSN, however, it is essential to consider the location and transmission time of the node due to the long propagation delay of acoustic medium. This problem is called spatial-temporal uncertainty. It can be defined as 'two-dimensional uncertainty', and is characterized as follows [31], [36]:

- The collision in the receiver is dependent on the propagation delay and transmission time, and can be explained as a duality that varies between both the transmission time and the location of the sensor nodes.
- The distance between the sensor nodes translates to uncertainty regarding the current channel status, and a packet may collide even if no other nodes send simultaneously.

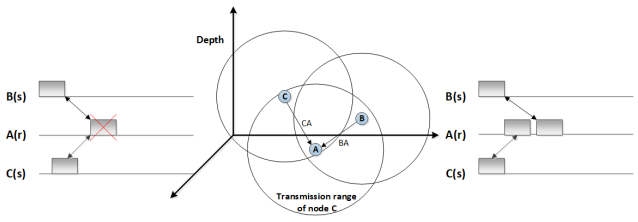


**FIGURE 1.** Impact of the long propagation delay on underwater MAC protocols.

Due to the possibility that a long propagation delay could cause a collision, two examples of the spatial-temporal uncertainty problem are presented in Figure 1. Firstly, when sensors *B* and *C* transmit packets with varying times of transmission, a collision might occur at sensor *A*, as shown on the left hand side in this figure. Secondly, when both sensors *B* and *C* start transmitting to sensor *A* simultaneously, there is no collision as their packets arrive at sensor *A* at different times, as illustrated on the right hand side of this figure. This is mainly because of various propagation delay incurred by each packet [42], [43].

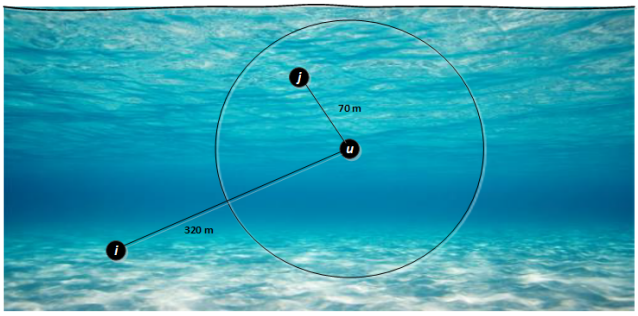


**FIGURE 2.** An example of near and far problem.

### B. NEAR AND FAR PROBLEM

Due to the unique characteristics of underwater acoustic channels, the near-far effect is mainly a major design challenge for MAC protocols [44], [45]. It is defined as being that when the received power for all nodes are not almost identical, signals from distant nodes cannot be received successfully. This requires that the transmission power of each sensor node must be controlled. As shown in Figure 2, the distance between *i* and *u* is significantly longer than the distance between *j* and *u*. As a result, the receiver node *u* receives different signal-to-noise ratio (SNR) levels of signals originating from each of the sender nodes.

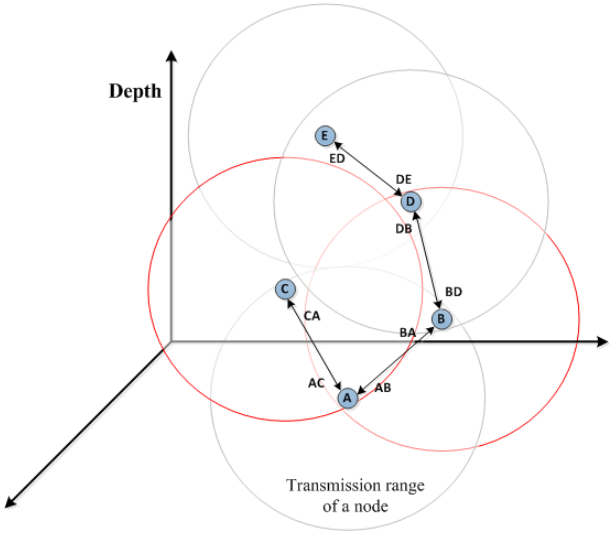


**FIGURE 3.** An example of hidden and exposed node problems.

### C. HIDDEN NODE PROBLEM

A hidden node problem can be defined as a sensor node which is not aware of another sensor. On-going transmission from both sensors may cause a collision at the destination sensor [46], [47]. This has been depicted in Figure 3, when sensors *B* and *C* are visible to *A*, but both of them (sensors *B* and *C*) cannot see each others. Consequently, transmitting packets from sensors *B* and *C* may result in a collision at sensor *A*. Moreover, the hidden node problem results in low throughput and high energy consumption.

### D. EXPOSED NODE PROBLEM

The exposed node problem occurs when a sensor node delay transmission because it overhears another transmission [21], [48]. This problem is shown in Figure 3, when sensors $B$ and $D$ are prevented from transmitting packets to their one-hop neighboring sensors $A$ and $E$ respectively. This is mainly because both sender sensors $B$ and $D$ are within each other transmission ranges, even though the receiver sensors $A$ and $E$ are out of each others transmission ranges, as shown in Figure 3. Specifically, if sensor $B$ transmits a packet to sensor $A$, sensor $D$ is prevented to transmit a packet to sensor $E$ after sensing the channel which might be interfered with the transmission by its one-hop neighbor sensor $B$. However, sensor $E$ still able to receive the transmission of sensor $D$ without interference.

## IV. GRAPH COLORING MAC PROTOCOL

This section presents the system assumptions used in our proposed GC-MAC protocol. It then gives an overview of our approach followed by how the network is modeled and also describing our GC-MAC protocol in detail.

### A. SYSTEM ASSUMPTIONS

An aquatic network is composed of several sensors uniformly scattered in an underwater area. A multi-hop acoustic network is considered so that sensors can achieve their respective destinations in a distributed manner using an omnidirectional and half-duplex acoustic modem.

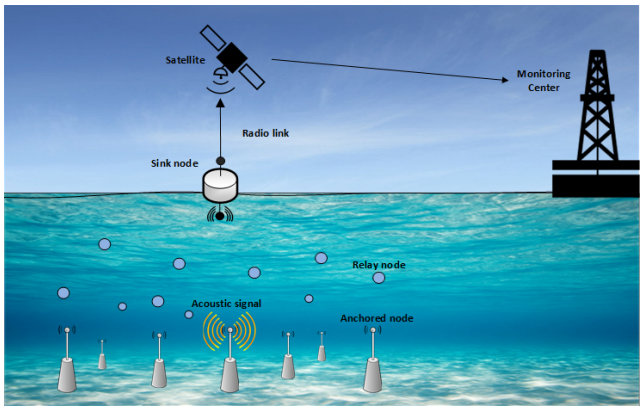


**FIGURE 4.** Network architecture.

In our three-dimensional (3D) underwater model, we assume that all sensors are randomly and uniformly distributed in an aquatic field. As demonstrated in Figure 4, a sink node, which is on the water surface, is equipped with an acoustic modem for underwater communication. It also uses a radio modem for out-of-water communication with the monitoring center. Anchored nodes are located at the bottom of the water in predetermined locations to collect information which is then delivered to the sink through relay nodes. The relay nodes are located at different depths between the anchored ones and the sink. Both anchored and relay nodes utilize acoustic signals to transmit data packets.

In this approach, we assume the acoustic transducers are set in an omnidirectional way. Every sensor is also assigned with a trio of coordinates $(x, y, z)$ to determine the distance between the sensors and their reference points, *rp*s. In addition, all sensors are homogeneous regarding to transmission range and energy consumption; and have unique identifier (ID).

### B. OVERVIEW

To improve the distributed MAC scheduling and to properly address the problems and challenges when designing our GC-MAC protocol, such as the spatial-temporal uncertainty, near-far effect, and hidden/exposed node problem, we use the concept of graph coloring approach to achieve better throughput, energy efficiency, and fairness in UWSNs. GC-MAC is inspired by graph coloring techniques to develop a reservation-based contention-free MAC protocol. This is achieved by using a distributed clustering approach for up to two-hop neighboring sensors and then to address the hidden and expose node problems by removing the possible color conflict in two-hop neighboring graph. Using a TDMA-like approach, GC-MAC is able to assign a time-slot, color, to every individual node in the network in a distributed manner. Nodes with the same colors can thus transmit concurrently without collision to support spatial reuse (i.e., Pconcurrent transmissions in different neighborhoods). The primary goal is to reduce energy consumption by assigning a unique time-slot to every node in a two-hop neighborhood graph. Sensor nodes are thus awake in some time-slots to send or receive packets and asleep in the remaining time-slots. GC-MAC trades off latency for high throughput, energy efficiency, and fairness, therefore, it is reliable and flexible to be used for different energy-critical applications in UWSNs.

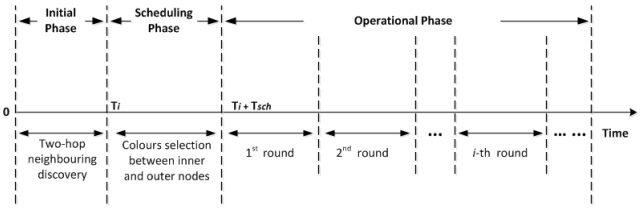


**FIGURE 5.** Timeline of GC-MAC protocol.

GC-MAC includes three phases to operate, which are initial, scheduling, and operational phase, as depicted in Figure 5. At the deployment process, the starting times of the initial and the scheduling phases for each sensor is set in order to let all sensors starting and ending each phase together. To avoid the impact of any clock skew that may occur over a long period of time, a guard time is applied. In addition, a summary of the notations used to describe the algorithms is given in Table. 1.

The purpose of the first phase, initial phase, is to collect information for two-hop neighboring sensors. This is achieved by exchanging a few small beacons. The length of

**TABLE 1. Notations used for explaining the GC-MAC algorithms.**

| Terms | Definition |
|---|---|
| $T_i$ | Initial phase interval (s) |
| $T_{sch}$ | Scheduling phase interval (s) |
| $T_{cd}$ | Conflict detection interval (s) |
| $CD$ | A delay timer used in the conflict detection interval (s) |
| $CH$ | A cluster head |
| $CL$ | A colouring list |
| $v$ | A sensor node |
| $v_i.Col$ | A node colour |
| $v_{ip}$ | A location of a node (inside or outside the internal cube) |
| $rp$ | A reference point |
| $d(v_i, rp_j)$ | A distance between a node and its reference point (m) |
| $F(ID, Col)$ | A set including the IDs and colours of neighbours |
| $p_c$ | A colouring packet |
| $p_{up}$ | An updating packet |
| $p_{out}$ | An outer updating packet |
| $N_g$ | Neighbouring graph |
| $Neig-info$ | A neighbouring information list |
| $msg_{CD}$ | A conflict detection message |

this phase, $T_i$, is a constant value set during the deployment time for all sensors.

The primary goal of the scheduling phase is to assign different colors to all sensors located within any two-hop neighborhood using a simple clustering approach. This is achieved by allowing a cluster head (CH), which is determined as the closest sensor to a reference point during the first phase, to assign a different color for all its one-hop neighboring (inner) sensors. Afterwards, the outer sensors, those located outside the cluster, decide about their own colors individually. By the end of the scheduling phase, every sensor node has a various color in any two-hop neighboring graph and hence no collision can occur. The length of the distributed scheduling phase, $T_{sch}$, is a predefined fixed value configured on each node prior to the deployment process. It should be long enough to allow sensors, from the seabed to the water surface, to reserve their time-slots but it is significantly shorter than that of the operational phase. Therefore, the energy consumed in the first two phases is negligible compared to the energy consumed in the operational phase. In terms of communication overhead, the initial and the scheduling phases only require to be repeated when the network topology changes.

The third phase, operational phase, is divided into a number of rounds. Each round consists of a number of time-slots. These time-slots are reserved after assigning a different color to each sensor. This means that every color represents a specific time-slot, as in conventional graph coloring, while the optimum spatial reuse can also be achieved by using the minimum number of colors. Sensors with the same color can transmit data packets at the same time without any collision while the hidden and exposed node problems are properly addressed. In this phase, the sensor nodes wake-up and sleep periodically. In other words, sensors can schedule to wake-up to send their own data packets during the reserved time-slots or to receive a data packet from a neighborhood, while they are asleep in other remaining slots when there is no data transmission or reception.

### C. NETWORK MODEL

To increase the efficiency of the distributed MAC scheduling, some reference points, *rp*s, are determined in which the locations of all *rp*s are well-known to all underwater sensors. Before the deployment process, the underwater network area is divided into a number of adjacent cubes to set a reference point at the center of every cube. Each cube is also including a smaller co-centered cube. Sensors are located inside the internal cube are called inner sensor nodes and the sensors are located outside of the internal cube, but still inside the external cube, are called outer sensor nodes. This is performed to classify the sensor nodes into two groups depending on their distances from the associated reference point.

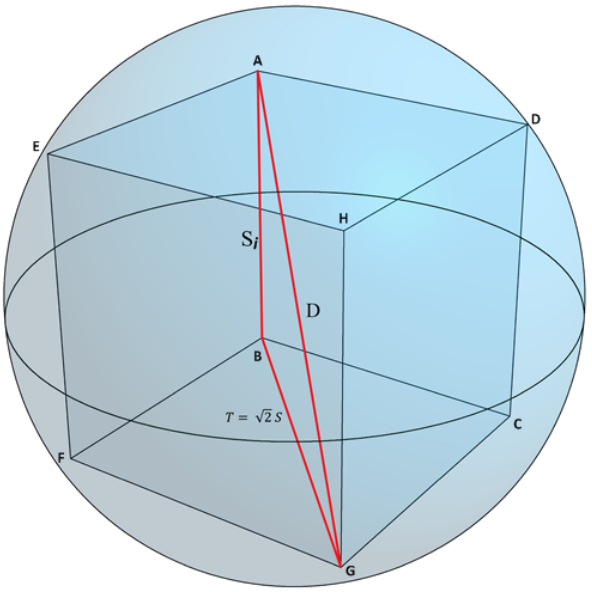


**FIGURE 6. Internal cube model of GC-MAC protocol.**

As shown in Figure 6, all vertices of the internal cube touch the surface of a sphere. The diameter of this sphere is the diagonal of the internal cube which is equal to $2 \times R_{tra}$, where $R_{tra}$ is the transmission range of a sensor node. Every edge of the internal cube, $S_i$, can be calculated as:

$$S_i = \frac{2 \times R_{tra}}{\sqrt{3}}. \tag{1}$$

The diameter of each face of the cube, $T$, can also be calculated by using the following equation:

$$T = \sqrt{2}\, S_i. \tag{2}$$

The distance between two sides of cubes is considered as $(R_{tra}/2)$ in each direction, as depicted in Figure 7. The edge of the external cube, $S_x$, is therefore given by:

$$S_x = S_i + R_{tra}. \tag{3}$$

Finally, the distance between adjacent reference points is considered as $S_i$ in each direction which is already calculated by Equation (1).

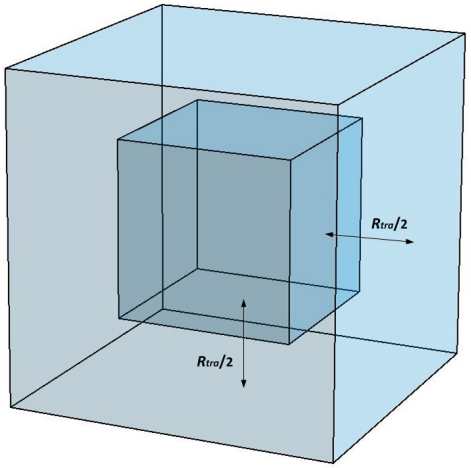


**FIGURE 7.** The position of the internal cube in relation to the external cube.

In our model, the distance between a node $v_i$ and its reference point $rp_j$ is defined by using the coordinate of node $v_i$ which is $(x_i, y_i, z_i)$ and the coordinate of the reference point $j$ $(rp_{j_x}, rp_{j_y}, rp_{j_z})$. Therefore, the distance between $v_i$ and $rp_j$ is given by:

$$d(v_i, rp_j) = \sqrt{(rp_{j_x} - x_i)^2 + (rp_{j_y} - y_i)^2 + (rp_{j_z} - z_i)^2}. \quad (4)$$

Let $v_{ip}$ determines a sensor node $v_i$ whether it is inside or outside the internal cube which can be computed by:

$$v_{ip} = \begin{cases} 1, & if \ \ | x_i - rp_x | \leq S_i/2, \\ & \quad | y_i - rp_y | \leq S_i/2, \\ & \quad | z_i - rp_z | \leq S_i/2 \\ 0, & otherwise, \end{cases} \quad (5)$$

where 1 means that a sensor node is located within the internal cube which identified as an inner node, and 0 denotes that a node is outside the internal cube. In other words, when the coordinate of node $v_i$ is smaller or equal than a half of the internal cube's side, $S_i/2$, it is considered as an inner sensor node, otherwise it is an outer sensor node.

Due to the different algorithms are used in the second phase, the inner and outer sensor nodes must be determined at the beginning of the scheduling phase.

### D. INITIAL PHASE

In the initial phase, two-hop neighboring information is obtained by performing two rounds of beaconing. Each sensor knows the start time of this phase during the deployment process.

In the first round, every sensor node broadcasts a beacon in a random time and it constructs a one-hop neighboring graph, $N_g$, by extracting the ID and distance to the reference point, $rp$, from all beacons received from neighboring sensors. To minimize the chance of collisions during this round, nodes randomly set a transmitting time for beaconing using timers when they broadcast beacon messages. This has been shown in Algorithm 1 between lines 1 and 10. When the beacon timer is expired, the sensor node creates a beacon message including the beacon type (*m.type*), sensor ID (*m.ID*), and its distance to the closest reference point (*m.d*), as shown in lines 4-7. Then, the beacon message is transmitted to the neighboring nodes and a new beacon timer is set for the sensor node (lines 8-9).

**Algorithm 1** One-Hop & Two-Hop Beaconing

```
1  Procedure Broadcast One-hop Info
2  // type = 0 means (1-hop beacon type)
3  m: a new beacon message
4  if beacon timer expired then
5      m.type ← 0
6      m.ID ← ID(v_i)
7      m.d ← d(v_i, rp_j)
8      Broadcast m
9      Set a new timer
10 end procedure
11 Procedure Broadcast Two-hop Info(N_g)
12 // type = 1 means (2-hop beacon type)
13 m: a new beacon message
14 if beacon timer expired then
15     m.type ← 1
16     m.ID ← ID(v_i)
17     m.N_g ← Neig−info(ID(v_i), d(v_i, rp_j))
18     Broadcast m
19 end procedure
```

In the second round, $N_g$ is broadcast by each sensor in a random time to be used for two-hop neighboring discovery. This procedure has been shown in Algorithm 1 (Lines 11-19). When the beacon timer of the second round is expired, the sensor node broadcasts a beacon message including the beacon type (*m.type*), sensor ID (*m.ID*), and the neighboring information list ($m.N_g$), as depicted in lines 14-18.

The length of this phase, $T_i$, is set to a predefined fixed value for all sensor nodes prior to the deployment process. This should be long enough to let all sensor nodes creating their own two-hop neighboring graphs with accurate information. However, the length of this phase is very short compared to that of the operational phase. To handle the fast topology changes in a highly mobile scenario, the total length of initial, scheduling and operational phases are considered shorter compared to a static scenario.

### E. SCHEDULING PHASE

This phase includes two rounds (inner nodes coloring and outer nodes coloring) to assign colors to all inner neighboring nodes and then to allow the outer nodes selecting their own colors individually.

#### 1) INNER NODES COLORING

During the first round, the cluster heads (CHs) are selected based on the closeness to a reference point (lines 2-5), which

**Algorithm 2** Inner Coloring & Updating

1 Procedure *Inner coloring*
2 $A \leftarrow \{\ a\ |is\ the\ node\ v_i\ with\ lowest\ distance\ to\ rp\}$
3 **if** $(v_i == A)$ **then**
4 $\quad F(ID, Col) = \phi$ // start with empty set
5 $\quad CH \leftarrow v_i$
6 $\quad p_c$: *a new coloring packet*
7 $\quad CH.Colour \leftarrow First-Colour$
8 $\quad F(ID, Col) \leftarrow F(ID, Col) \cup Colour(CH)$
9 $\quad$ **for** $n \in neighbors(CH)$ **do**
10 $\quad\quad n.Colour \leftarrow Colour-Selection(n)$
11 $\quad\quad F(ID, Col) \leftarrow F(ID, Col) \cup Colour(n)$
12 $\quad p_c.CL(ID, Col) \leftarrow F(ID, Col)$
13 $\quad$ Broadcast $p_c$
14 end procedure
15 Procedure *Inner updating*$(p_c)$
16 **if** *node* $\in p_c.CL(ID, Col)$ **then**
17 $\quad node.Colour \leftarrow$ Get $Colour$ from $p_c.CL(ID, Col)$
18 $\quad p_{up}$: *a new updating packet*
19 $\quad F(ID, Col) = p_c.CL(ID, Col)$
20 $\quad$ **for** $n \notin neighbors(node)$ **do**
21 $\quad\quad F(ID, Col) \leftarrow F(ID, Col) - n(ID, Col)$
22 $\quad p_{up}.CL(ID, Col) \leftarrow F(ID, Col)$
23 $\quad$ Broadcast $p_{up}$
24 end procedure

is known by all sensor nodes in every two-hop neighborhood after the initial phase. Each CH individually chooses a color for itself and assign different colors to its inner nodes cluster members (CMs) by broadcasting a coloring packet, $p_c$, as shown in lines 7-13. This packet includes the IDs and colors for itself and all inner nodes located into the internal cube. Hence, every sensor node located into the internal cube, inner node, has a different color in any two-hop neighboring graph. The network density can determine the total number of colors used in the scheduling phase.

Each inner node obtains its specific color by receiving the coloring packet, $p_c$, from the CH and it then broadcasts another packet called updating packet, $p_{up}$, containing its ID, color, neighbors' IDs, and neighbors' colors, to the outer neighboring nodes, those are located outside of the internal cube, but still inside the external cube (lines 16-23). By receiving $p_{up}$, the outer nodes are updated about the colors of the neighboring inner nodes. The procedure of inner node coloring and outer nodes updating is shown in Algorithm 2.

#### 2) OUTER NODES COLORING

During the second round, every outer node checks its inner neighboring nodes along with the attached coloring list, $CL$, from the received updating packet, $p_{up}$, in order to remove non-neighboring nodes (lines 2-3). Thereafter, each outer node forwards it as a new packet called outer coloring packet,

**Algorithm 3** Outer Coloring

1 Procedure *Outer coloring*$(p_{up})$
2 **if** *node* $v_i$ *receives* $p_{up}$ ***and*** $v_{ip} = 0$ **then**
3 $\quad$ Update $N_g$ by $p_{up}.CL(ID, Col)$
4 $\quad p_{out}$: *a new outer updating packet*
5 $\quad$ Add neighbors with colors from $N_g$ to $F(ID, Col)$
6 $\quad p_{out}.CL(ID, Col) \leftarrow F(ID, Col)$
7 $\quad$ Broadcast $p_{out}$
8 $\quad$ **if** $N_g$ *is fully updated by all neighbors colors* **then**
9 $\quad\quad$ node $v_i$ selects its color based on the ID priority among neighbors
10 end procedure

$p_{out}$, to all other neighboring nodes (lines 4-7). Now, every outer node is able to select a color that is different than any two-hop neighboring nodes. The priority of color selection between the outer nodes is set based on the lower ID. In other words, the lower the outer node's ID, the higher the priority to select the first available color (lines 8-9). It should be noted that the ID of the sensor nodes has already been exchanged during the first phase. Algorithm 3 describes how the priority between the outer neighboring nodes is exchanged. The length of the scheduling phase, $T_{sch}$, is a constant value set during the deployment process for all sensor nodes.

### F. OPERATIONAL PHASE

The operational phase consists of a number of rounds, each round including a number of time-slots. In every round, sensors are awake in some time-slots to send or receive data packets and asleep over the remaining time-slots. Specifically, all the time-slots are represented in different colors to each. In other words, every color represents a different channel reservation as in conventional graph coloring, while the spatial reuse can optimally be achieved by using the minimum number of colors. Every sensor node is, therefore, able to reserve a time-slot, color, in any two-hop neighborhood in a distributed manner, and aware of the time-slots assigned by its neighborhood. Sensors with the same color are in the transmission mode to transmit their data packets or in the listening mode to receive the data packets from their neighboring nodes. They are in the sleeping mode during the remaining time-slots, if there is no data transmission and reception scheduled among them. In this way, our proposed protocol ensures collision-free scheduling by assigning a different slot, color, to each sensor in every two-hop neighborhood. Sensors with the same color can concurrently transmit data packets with no chance of collisions. This method is repeated in each round. The length of the operational phase (i.e., the number of rounds) depends on the topology changes due to energy depletion or node displacement. For highly mobile scenarios with rapid topology changes, the length of this phase should be shorter compared with stationary or limited mobile nodes.

In either way, it is a predefined fixed value configured on each sensor node before deployment.

The duration of every round time, $T_r$, has a reverse relationship with traffic rate, $\lambda$, which is presented in terms of packets per second. The higher the traffic rate, the shorter round time and, hence, the shorter the sleeping time. The length of each round, $T_r$, is given by:

$$T_r = \frac{1}{\lambda}. \tag{6}$$

Every $T_r$ is divided into a number of equal size time-slots, $N_s$. The number of time-slots per round depends on the round time, $T_r$, and time-slot lengths, $T_s$, which can be calculated by using the following equation:

$$N_s = \frac{T_r}{T_s}, \tag{7}$$

where $T_s$ is the length of every time-slot, which is longer than the propagation delay to ensure that a packet is entirely received at the destination before starting of data transmission by another node. The length of each time-slot, $T_s$, is given by:

$$T_s = T_d + T_{guard}, \tag{8}$$

where $T_{guard}$ indicates the guard time and $T_d$ denotes the propagation delay of a transmitted packet which can also be calculated by using the following equation:

$$T_d = \frac{R_{tra}}{V_s}, \tag{9}$$

where $R_{tra}$ is the transmission range of each sensor node, and $V_s$ is the velocity of sound in water.

However, the traffic rate, $\lambda$, used in Equation (6) should be limited to be used in our system model. To find its upper-bound, the following equation must be satisfied:

$$T_s \leq T_r, \tag{10}$$

where the time-slot length, $T_s$, cannot be exceeded the round time, $T_r$. The time-slot length, $T_s$, is already calculated by Equation (8). By replacing $T_r$ using Equation (6), it can be extended as:

$$T_s \leq \frac{1}{\lambda}. \tag{11}$$

Based on the above equation, the upper-bound for $\lambda$ is calculated as:

$$\lambda \leq \frac{1}{T_d + T_{guard}}. \tag{12}$$

The upper-bound for $\lambda$ depends on the length of slot. The length of time-slot is a fixed value which is long enough to handle the consecutive receiving packets.

According to Equation (6), the length of the round time, $T_r$, is decreased by increasing the traffic rate. In turn, the lower $T_r$ implies less number of time-slots per round according to Equation (7). In other words, when $T_r$ is low, GC-MAC allows sensor nodes to be in the sleeping mode for a longer period of time. However, when $T_r$ is high, all sensor nodes are active most of the time to transmit or receive data packets.

## V. TWO SPORADIC COLLISION SCENARIOS

By broadcasting two hops coloring information, the hidden node problem can be addressed in almost all scenarios which makes our proposed algorithm a collision-free protocol. However, there are only two very specific scenarios where the same color might be selected by two hidden nodes located in the two-hop graph of a node. This is mainly because the colors of inner nodes have already been given by their CHs, which only causes a rare conflict scenario as depicted in Figure 8.

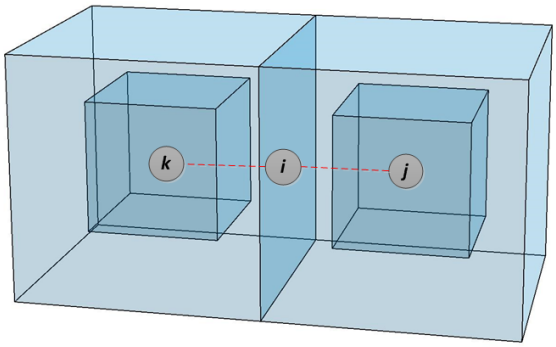


**FIGURE 8.** A very specific scenario may lead to a potential collision, which has already been handled in GC-MAC.

Since inner nodes $k$ and $j$ are in two separate decision areas, both may obtain, with a very slim chance, the same scheduling color by their own cluster head nodes independently. The same slot is therefore reserved to transmit their data packets. Given that nodes $j$ and $k$ distances to $i$ is close to each other and both have a packet to transmit, a collision may occur at node $i$.

As another potential case, two outer nodes located in two different sides of the internal cube and are neighboring nodes with an inner node, may independently select the same color. In this case, transmitting a packet by both outer nodes to an inner node results in a conflict. This scenario has been illustrated in Figure 9, where sensors $m$ and $p$ cannot hear from one another. Due to sensor $p$ is unable to know the time-slot, color, reserved by sensor $m$, sensor $p$ may obtain the same time-slot, color. Hence, sending a packet by sensors $p$ and $m$ may cause a collision in sensor $n$.

To address these issues, the concept of conflict detection (CD) is introduced here, as depicted in Algorithm 4. Since $C_{cl}$ is the conflict color reserved whether by two separate inner nodes within two different internal areas as shown in Figure 8, or reserved by two outer nodes within an external area as specified in Figure 9, thus transmissions from both $v_{s1}$ and $v_{s2}$ nodes to $v_r$ may cause collision. In $C_{cl}$, a conflict slot could be caused by inner or outer nodes as a reception slot (RS). To be collision-free, both $v_{s1}$ and $v_{s2}$ transmissions to $v_r$ cannot overlap within the same RS. To do so, the conflict detection period, CD, is added at the end of scheduling phase. The interval of the conflict detection, CD, should be long enough to ensure that a conflict detection, which is caused by those specific circumstances,

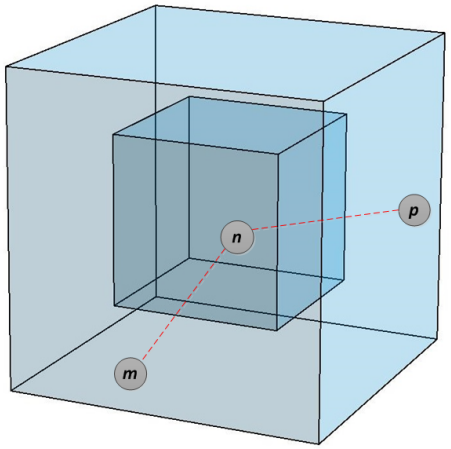


**FIGURE 9.** Another specific scenario leads to a possible collision. This has also been addressed in GC-MAC.

**Algorithm 4** Conflict Detection (CD)

Procedure *Conflict detection*(*node*)
**if** $v_i$, $v_j \in$ *neighbors (node)* ***and*** $v_i.Col == v_j.Col$ **then**
    node selects a delay timer, CD, randomly within the conflict detection interval $[0, T_{cd}]$
**while** *CD is not expired* **do**
    node listens to $msg_{CD}$ from neighboring nodes
    **if** *conflict is fixed* **then**
        Update $v_i$ and $v_j$ in $N_g$
**if** *CD is expired* ***and*** *conflict is not yet fixed* **then**
    node sends a $msg_{CD}$ to $v_i$, $(ID.v_i > ID.v_j)$
    $v_i.Col$ is updated and sent to neighboring nodes
end procedure

is entirely resolved and rescheduled. According to the above specific scenarios, a node, between two hidden nodes whether its considered outer or inner sensor node as shown in Figure 8 or Figure 9 respectively, is responsible to fairly reschedule the conflict between them by sending a conflict detection message, $msg_{CD}$, to the node that has a higher ID than another. This message includes a new updated color. Upon receiving the conflict detection message $msg_{CD}$, a node requires to inform its neighboring nodes about the updated color. However, the length of conflict detection, CD, is considered as, $T_{cd}$, which is very light compared with the scheduling phase, $T_{sch}$.

Figure 10 shows an example of different assignment algorithms to schedule inner and outer sensor nodes with different colors in any two-hop neighboring graph. To overcome hidden and exposed node problems, two-hop beaconing information is used, by which the inner and outer sensor nodes are also determined. Inside the internal area, the node with the lowest distance to the reference point individually becomes the cluster head, CH, of the sub-graph. In this figure, every CH independently selects the first color (assuming it is red)

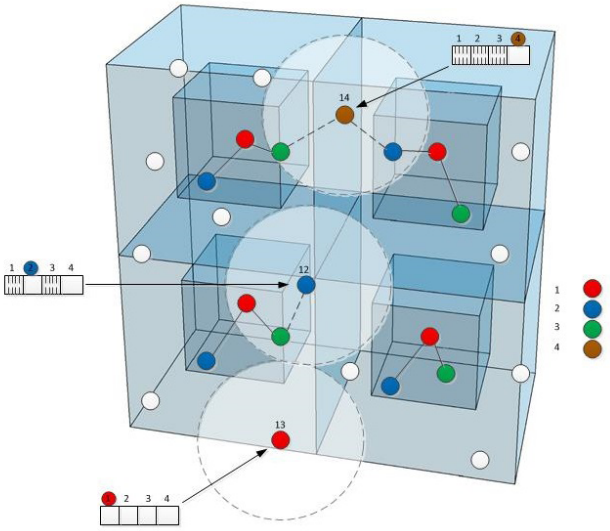


**FIGURE 10.** Distributed disjoint two-hop sub-graph.

for itself and provides different colors for its inner neighboring nodes within two-hop neighboring graph. A coloring packet is exchanged to all inner nodes to find their colors and then every inner node informs its outer neighboring nodes about its color and neighboring nodes' colors by sending another packet called updating packet, $p_{up}$. Afterwards, every outer node checks the coloring of its inner neighboring nodes from the received $p_{up}$ in order to remove non-neighboring nodes. Each outer node then forwards that information within a new packet called outer coloring packet, $p_{out}$, to all other neighboring nodes. Every outer node is, therefore, able to select a color that is different than any two-hop neighboring nodes. This is achieved when the outer node hears from all other outer neighboring nodes, it can select its color based on the node's ID priority (lower ID).

To illustrate the priority of selecting colors among outer sensor nodes considering the topology shown in Figure 10. Every outer sensor represents its ID (on the top). The lower the ID, the higher the priority to obtain the first available color. The priority thus has reverse proportion relation with the lowest ID of sensor nodes. In this figure, some specific scenarios are illustrated. The node with the lowest ID (12) has only a green inner neighboring color. In this case, this node obtains a blue color based on its slot availability as well as selecting a color within a two-hop graph. In the second scenario, the next lowest ID (13) obtains the first color (red). This is because this node has no either inner or outer neighboring nodes, therefore it is prioritized to obtain the first color. In the final scenario, the node with ID 14 receives updating packets from both green and blue neighboring colors. Both nodes are located into two different internal areas with two different cluster heads. A new color is thus generated for this node in which the number of colors is depending on the numbers of sensor nodes that are deployed in the underwater area, as previously mentioned. Note that the fewer colors used,

the higher the spatial reuse will be; consequently, the lower the channel utilization between vertices.

## VI. COVERAGE AND CONNECTIVITY ANALYSIS

The three-dimensional (3D) space is surprisingly difficult to optimality cover. Comparing with 2D, it has a finite number of faces, edges, and vertices. The coverage and connectivity in this area has been widely studied [49], [50]. In this section, the volumetric quotient of our selected shape (cube) is calculated and then compared with another shape (square pyramid) to show the effectiveness of cubes to cover the 3D underwater environment. Using the cubes can ensure a space-filling without any overlap or gap. This is examined by presenting a comparison study of the sensors density in each shape.

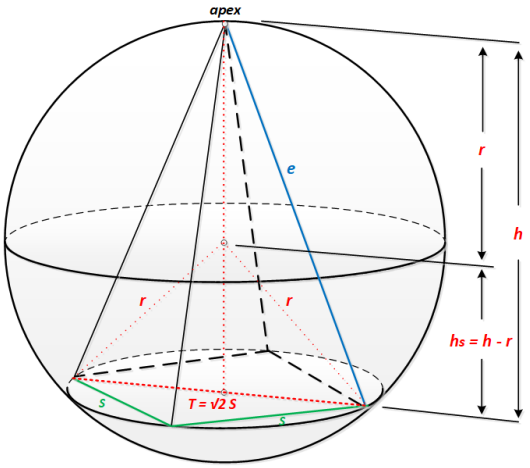


**FIGURE 11.** A square pyramid with five faces and vertices, and eight edges.

### 1) SQUARE PYRAMID

The volumetric quotient of a square pyramid relies on its height, $h$, and base side, $S_i$. We assume all vertices of the square pyramid touch the surface of a sphere. The length of the base side, $S_i$, is equal to $(2r/\sqrt{3})$ and the volume of a sphere is considered to be $(\frac{4}{3}\pi r^3)$, where $r$ represents the radius of its circumsphere which is exactly equal to the transmission range of each sensor ($R_{tra} = \sqrt{3}\ S_i\ /2$). Thus, the $h$ of the square pyramid (distance from the apex, which is perpendicularly above the center of the square, to the center of the pyramid's base, as depicted in Figure 11) is given by:

$$h = r + h_s. \tag{13}$$

In order to calculate $h$ based on $r$, we need to calculate $h_s$ which is given by:

$$h_s^2 = r^2 - (T/2)^2, \tag{14}$$

by replacing $T$ using Equation (2):

$$h_s^2 = r^2 - (\frac{\sqrt{2}S_i}{2})^2, \tag{15}$$

by substituting $S$ using Equation (1):

$$h_s^2 = r^2 - (\frac{\sqrt{2}r}{\sqrt{3}})^2, \tag{16}$$

thus, $h_s$ can be calculated based on $r$ as the following:

$$h_s = \frac{r}{\sqrt{3}}. \tag{17}$$

Therefore, by replacing $h_s$ in Equation (13), $h$ can be computed as:

$$h = r + (\frac{r}{\sqrt{3}}). \tag{18}$$

The volume of the square pyramid is $(\frac{1}{3} \times S_i^2 \times h)$ and the volumetric quotient is given by:

$$\frac{1}{3} \times S_i^2 \times h \Big/ \frac{4}{3}\pi r^3 = \frac{h \times S_i^2}{4\pi r^3}, \tag{19}$$

by replacing $S_i$ and $h$ using Equations (1) and (18) respectively, the volumetric quotient of a pyramid is:

$$\begin{aligned} \frac{(r + \frac{r}{\sqrt{3}}) \times (\frac{2r}{\sqrt{3}})^2}{4\pi r^3} &= \frac{(r + \frac{r}{\sqrt{3}}) \times 4r^2}{3 \times 4\pi r^3} \\ &= \frac{(r + \frac{r}{\sqrt{3}})}{3\pi r} = \frac{r(1 + \frac{1}{\sqrt{3}})}{3\pi r} \\ &= \frac{(1 + \frac{1}{\sqrt{3}})}{3\pi} = 0.16735 \end{aligned} \tag{20}$$

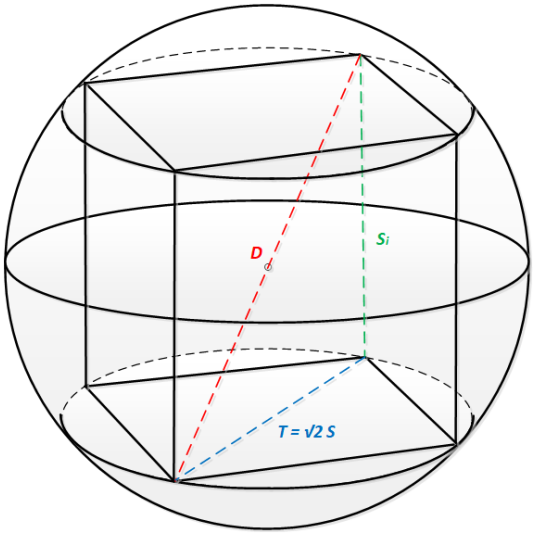


**FIGURE 12.** A cube with six faces, eight vertices, and twelve edges.

### 2) CUBE

The volume of a cube is equal to $(S_i^3)$, where $S_i$ is the length of the cube's side, and the volume of a sphere is already known as $(\frac{4}{3}\pi r^3)$. If all vertices of the cube touch the surface of the sphere, then the $r$ of its circumsphere is $(\sqrt{3}S_i/2)$ or equal to $R_{tra}$ which is the transmission range of each sensor. The diameter of the sphere, $D$, is the diagonal of the cube which is equal to $(2 \times r)$, as depicted in Figure 12. The length of the

cube's side is already calculated in Equation (1). Therefore the volumetric quotient of a cube can be computed by:

$$S_i^3 \Big/ \frac{4}{3}\pi r^3, \tag{21}$$

by substituting $r$, it can be extended as:

$$S_i^3 \Big/ \frac{4}{3}\pi \left(\frac{\sqrt{3}}{2}S_i\right)^3 = \frac{2}{\pi\sqrt{3}} = 0.36755 \tag{22}$$

#### 3) COMPARISON

Among these shapes, the cube shows a better volumetric quotient than that of the square pyramid. In addition, we compare the number of sensors that is required by each shape. The number of sensor nodes that eligible to use by the square pyramid is 0.36755/0.16735 = 2.196 times that of the cube. Table 2 summarizes the results.

**TABLE 2. Volumetric quotient of two types of space-filling polyhedrons.**

| Polyhedron | Volumetric quotient | Number of nodes needed compared to the square pyramid |
|---|---|---|
| Square pyramid | 0.16735 | same |
| Cube | 0.36755 | 54.47% less |

As shown in Table 2, the cube requires less number of nodes compared to the square pyramid by 54.47%. On the contrary, the square pyramid needs more number of nodes compared to the cube by 119.62%. By selecting the cube as the best volumetric quotient, our graph coloring model is fully covered without any gap and its scheduled guarantees no chance of overlapping that may occur between nodes, as explained earlier in Section [IV-C].

## VII. PERFORMANCE EVALUATION

This section first discusses the simulation setup of the proposed GC-MAC protocol in the Aqua-Sim underwater simulation [51]. It then defines the important medium access metrics used in our performance study. It also proceeds to evaluate GC-MAC performance and to compare it with ED-MAC [33], [34], T-Lohi [52], and UWAN-MAC [10] protocols through simulation.

### A. SIMULATION SETUP

This study implemented GC-MAC in Aqua-Sim, an NS-2 based simulator for underwater sensor networks. Simulations were performed with the following parameters, unless otherwise noted. The power consumption on the transmission, reception, and sleeping modes are 2 Watts, 0.75 Watts, and 8 mW, respectively. The length of the data packet is set to 1000 bits, and the length of all other control packets are set to 100 bits. The channel bit rate (i.e., bandwidth) is set 10 kb/s and the maximum transmission range is 100 m. The simulation time is set to 30 minutes.

In our simulation, we consider two different network scenarios by changing traffic rates and numbers of nodes. We randomly deploy a number of sensors in a 3D region of $216m \times 432m \times 432m$ for a fully connected network. The input parameters to evaluate our model are summarized in Table 3.

**TABLE 3. Input parameters.**

| Parameter | Value |
|---|---|
| Acoustic propagation speed | 1500 m/s |
| Bandwidth | 10 Kb/s |
| Control packet size | 100 bits |
| Data packet size | 1000 bits |
| Deployment region | 216 m × 432 m × 432 m |
| Idle power | 8 mW |
| Length of the initial phase | 60 s |
| Length of the scheduling phase | 120 s |
| Maximum transmission rage | 100 m |
| Movement model | RandomWalk 2D mobility model |
| Movement speed of nodes | 2 m/s, change movement direction every 2 s |
| Node Number | 50–500 sensor nodes |
| Offered load | 1–8 packet/s |
| Receiver power | 0.75 Watts |
| Running rounds | 20 |
| Simulation time of one round | 1800 s |
| Traffic rate | 0.05–0.5 packet/s |
| Transmission power | 2 Watts |

In the first scenario, we distributed 50 sensors into the given area to compare the packet delivery ratio, throughput, energy consumption, and fairness index of GC-MAC, ED-MAC, T-Lohi, and UWAN-MAC protocols with varying traffic rates.

In the second scenario, we vary the network density. In this case, the traffic rate is kept fixed to 0.1 packets per second and the number of nodes increases from 50 to 500 nodes.

In our simulation setup, we consider $T_i$ as 60 seconds and $T_{sch}$ as 90 seconds. The duration time of conflict detection, $T_{cd}$, is considered as 30 seconds, which is added to the scheduling phase, $T_{sch}$. Thus, the total length of $T_{sch}$, including the $T_{cd}$, sets to 120 seconds.

### B. PERFORMANCE METRICS

We define a number of metrics to compare the performance of GC-MAC protocol with that of ED-MAC, T-Lohi, and UWAN-MAC protocols; namely, packet delivery ratio (PDR), throughput, energy consumption, and fairness index as functions of traffic rates and numbers of nodes.

The packet delivery ratio, PDR, is defined as the ratio of the packets successfully received in relation to the total packets generated in the network. The network throughput is defined as the number of data packets per second that successfully arrive at their destinations. Energy consumption is defined by dividing the total energy usage in the network by the successfully delivered data packets, where the energy consumption is measured in joules per packet.

The fairness index is the most important MAC protocol metric in order to evaluate its performance regarding the equality of sensors to access the medium channel. To measure the fairness index of the above-mentioned protocols, we adopt

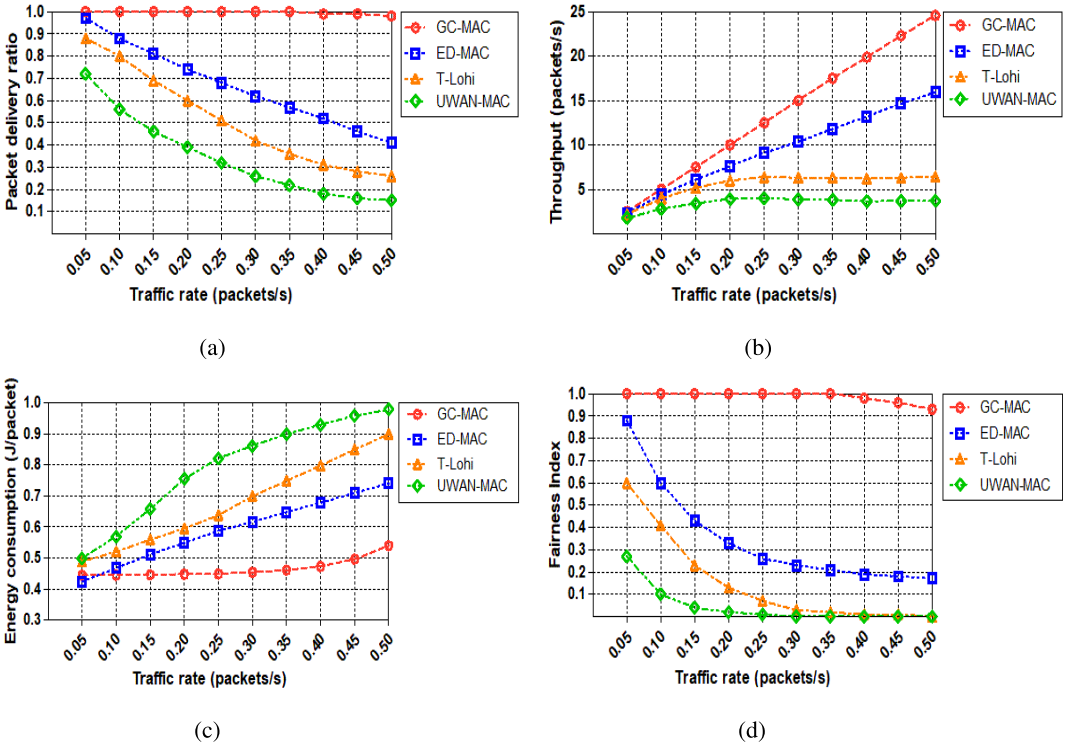


**FIGURE 13.** PDR, Throughput, Energy consumption, and Fairness index vs. Traffic rate. (a) PDR. (b) Throughput. (c) Energy consumption. (d) Fairness index.

the Jain's Fairness Index [53]:

$$\text{Fairness Index} = \frac{(\sum x_i)^2}{(n.\sum x_i^2)}, \tag{23}$$

where $x_i$ is the throughput of a sensor node $i$, $1 <= i <= n$, and $n$ is the number of nodes in the network. The fairness index can be measured based on the value ranges between 0 and 1. The higher the value, the fairer to access the medium channel, i.e., when the value becomes closer to 1, the fairness index increases. When this value closer to 0, the fairness index decreases. This means that accessing the medium channel is fully fair when the fairness equals 1.

### C. SIMULATION RESULTS

Our proposed protocol (GC-MAC) is compared against other contention-based and contention-free MAC protocols through simulations. For each test, the results are averaged over 20 runs, with a randomly generated topology in each run.

Figure. 13 shows how the load affects the performance of each protocol. Figure (13a) shows the PDR as a function of traffic rate. Our proposed GC-MAC protocol delivers all the data packets successfully until 0.35 packets/s, and then its PDR slightly dropped when the traffic rate further increases. GC-MAC with conflict detection scheme hence achieves a higher performance than other protocols. ED-MAC also achieves a high performance at low traffic rates. When the traffic is higher than 0.15 packets/s, however, the PDR of ED-MAC protocol is significantly decreased.

As shown in Figure (13b), the network throughput of all protocols is proportional to the traffic rates. When the traffic rate increases, the network throughput increases correspondingly, except for T-Lohi and UWAN-MAC, which reach a saturation point. Our proposed protocol, GC-MAC, achieves a higher network throughput than that of ED-MAC, T-Lohi, and UWAN-MAC when their traffic rates are the same. This is because GC-MAC employs an effective conflict detection algorithm to avoid collisions, thus improving the network throughput considerably. As the traffic rate increases, ED-MAC's throughput decreases significantly even it is almost considered as collision-free approach, but there are still specific circumstances that lead to collisions.

Figure (13c) dramatically demonstrates that our collision-free protocol is much more energy efficient than other protocols in this simulation as a function of traffic rate. On the one hand, as the traffic rate increases, wasted energy in GC-MAC greatly stable. This is mainly because it prevents any possibility of collisions by applying a conflict detection mechanism, which highly conserves energy. On the other hand, when the traffic rate exceeds 0.35 packet/s, its energy consumption slightly increases to achieve just over 0.5 joules per packet at 0.5 packet/s of the traffic rate. ED-MAC, however, when used at a lower traffic rate, consumes less energy than GC-MAC, T-Lohi, and UWAN-MAC. When the traffic rate increases

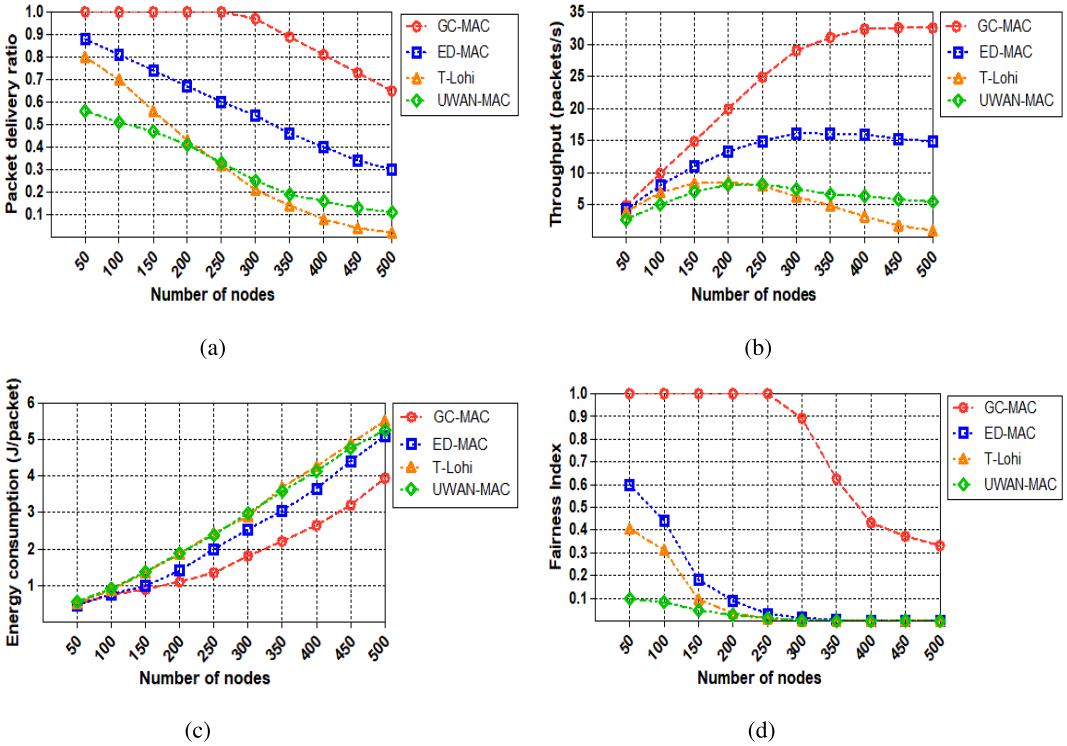


**FIGURE 14.** PDR, Throughput, Energy consumption, and Fairness index vs. Number of nodes. (a) PDR. (b) Throughput. (c) Energy consumption. (d) Fairness index.

further, the ED-MAC energy cost increases considerably, by almost 19.6%, which is more than that of GC-MAC due to potential collisions caused by two hidden nodes which are neighbors of another node at a lower depth. Another reason is that the traffic rate and the round time duration have a reverse relationship, i.e., when the traffic rate further increases, the duration of round time decreases. As a result, some sensor nodes cannot obtain a free time-slots. T-Lohi, on the other hand, shows a rapid rise in energy consumption per packet due to an increasing number of collisions of control packets, while UWAN-MAC consumes much more energy than all other protocols during all traffic rates. This is due to the inefficient scheduling of UWAN-MAC, which causes considerably more collisions and retransmissions.

Figure (13d) shows the result of an experimental set-up consisting of 50 nodes that are run for half an hour to strenuously test the fairness index of all mentioned protocols. First, it is observed that our proposed GC-MAC protocol with the conflict detection concept exhibits a high fairness index, achieving 100%, then maintaining the same percentage until 0.35 packets/s of the traffic load. When the traffic rate further increases, however, its fairness index slightly decreases. The proportional fairness considered in GC-MAC allocates the channel access according to the color assignment within a two-hop distance in order to avoid any possibility of collisions, as well as balancing the throughput and fairness among competing senders. Secondly, it is also observed that ED-MAC almost achieves the same fairness index and packet delivery rate, PDR, of GC-MAC at a low traffic rate. This is due to the slot length and duty cycle being calculated differently depending on their scheduling strategies. Since the traffic load is above 0.15 packets/s, the fairness index of GC-MAC outperforms that of ED-MAC by 2%. Another reason of that both protocols serve and improve the temporal and spatial reuse efficiency. These techniques are not applicable with T-Lohi and UWAN-MAC protocols, therefore their fairness index considerably decreases as the traffic rate increases.

Figure (14) shows how sparse and dense nodes can affect the performance of the GC-MAC, ED-MAC, T-Lohi, and UWAN-MAC protocols in order to study the network scalability and flexibility offered by each protocol. In this set of simulations, we keep the traffic rate fixed at 0.1 packets per second.

Figure (14a) shows that the PDR decreases as the node number increases, except in GC-MAC. This is because of the efficient scheduling as well as the conflict detection technique. With 50 nodes, the PDR of ED-MAC achieves nearly 88% compared to 80% and 56% for T-Lohi and UWAN-MAC respectively. This is due to the specific benefits of both GC-MAC and ED-MAC, such as high scalable scheduling; consequently they can handle more packets than either T-Lohi or UWAN-MAC. Among these, still there are very specific circumstances that may cause collisions, however, as stated in [33] and [34]. When the number of nodes gradually begins

to increase, the PDR of ED-MAC begins to rapidly decrease. This is mainly because ED-MAC schedules based on its depth criteria, whereas GC-MAC schedules based on clustering and graph coloring approaches. It can be observed that our GC-MAC achieves higher PDR than all other protocols in all sparse and dense scenarios. This is due to its collision avoidance capabilities. Moreover, as the node density increases, T-Lohi's PDR rapidly drops because of the increasing of tone packet collisions.

As shown in Figure (14b), the network throughput of all MAC protocols is drawn as a function of the numbers of nodes. In this figure, an interesting phenomenon can be observed, which is that all protocols (within a 50-node density) can almost deliver the same amount of data packets successfully. As the number of nodes increases, the network throughput increases correspondingly and eventually reaches a saturation point, except for GC-MAC, with a continuously rising throughput of 33 packets per second. The GC-MAC protocol outperforms other protocols in the same circumstances. This is due to its specific benefits, such as highly scalable scheduling and its conflict detection technique which enables it to handle more data packets than other contention-free and contention-based protocols, as well as to solve any conflict among sensor nodes. Moreover, we can see that the throughput of T-Lohi reduces in a denser network. This is because there is more intensive competition between nodes to reserve the contention round (CR).

In Figure (14c), the energy consumption of all protocols is inversely proportional to the number of nodes. As the number of nodes increases, the energy consumption increases correspondingly. This is because when the number of nodes increases, more nodes are involved, therefore there is more intensive competition to access the channel. GC-MAC consumes the least energy out of all the protocols, because it adopts energy conservation measures by considering the collision avoidance algorithm. This consideration makes collisions not able to occur, thus GC-MAC reduces energy consumption as well as improving throughput and channel utilization. T-Lohi consumes the highest energy among all protocols in that it does not adopt any energy conservation measures when the node density increases. This is also because when the node density increases, more contention rounds, CRs, are needed for contending the channel; the number of collisions of tone packets thus also increases. Specifically, our proposed protocol consumes 20% less energy than ED-MAC and 24% less than the energy consumption of T-Lohi in high density. There is, therefore, a large gap between GC-MAC and other protocols, which demonstrates that GC-MAC can perform highly in a distributed manner.

In Figure (14d), the simulation results show that our proposed scheme (GC-MAC) achieves 100% of its fairness index even if the density is increased up to 250 nodes. When the node density further increases, however, our scheme protocol's fairness index sharply decreases. This is mainly because the network congestion reduces the fairness index in all protocols. Due to the large delays in underwater acoustic networks, the distance between nodes becomes a key factor in competitive channels, therefore more intensive competition to reserve the channel is observed. ED-MAC achieves lower fairness index than that of GC-MAC by 40% with 50 nodes. When the node density increases, its fairness index decreases considerably. The latter protocol achieves a better fairness index than that of ED-MAC when the node density is increased, however. Due to the lack of any capability for collision avoidance, the fairness index of ED-MAC significantly drops by more than double that of GC-MAC. A similar phenomenon is also observed in the cases of T-Lohi and UWAN-MAC. This is because neither protocol can avoid hidden node problems, which lead to increase in the number of collisions and retransmissions. Moreover, UWAN-MAC involves unknown propagation delays, which affects its fairness compared to other protocols.

## VIII. CONCLUSION

Designing medium access control (MAC) protocols for UWSNs is a critical task, compared to that of terrestrial sensor networks, mainly due to some unique features of acoustic signals in underwater environment. In this paper, we have proposed an efficient collision-free graph coloring MAC protocol (GC-MAC) for underwater sensor networks. GC-MAC generally uses a graph coloring approach with a duty cycle mechanism in order to assign a unique timeslot, color, to each node in every two-hop neighborhood. Hence, nodes with the same colors can transmit concurrently without any collision. Using our proposed protocol, nodes are awake in some slots to transmit or receive data and asleep over the remaining slots. In our proposed protocol, the near-far effect, spatial-temporal uncertainty, and hidden/exposed node problems have been addressed. It also detects two potential scenarios, under very specific circumstances. These two specific scenarios have been resolved by applying conflict detection algorithm. Hence, GC-MAC guarantees collision-free transmissions for data packets through the scheduling phase. Furthermore, it does not require special modem capabilities such as CDMA, and therefore the protocol is more likely to work with most contemporary modems. Using an extensive simulation study, the performance of GC-MAC has been compared against those of contention-free (ED-MAC) and contention-based (T-Lohi and UWAN-MAC) protocols. Simulation results have shown that GC-MAC outperforms those three MAC protocols in terms of packet deliver ratio, throughput, energy consumption, and fairness index with varying traffic rates and numbers of nodes.

As a future work, that would be interesting to improve and enhance the channel utilization by assigning the required slots adaptively, instead of considering a fixed number of slots per neighborhood. Another future interest lies in testing the reliability of the protocol in case of using the mobile Autonomous Underwater Vehicle (AUV) in a distributed manner in order to design an efficient MAC protocol along with data-gathering schemes.

## REFERENCES


[1] S. Xiong, C. Yuan, L. Tian, and Y. Zhan, "RET-MAC: A new fair MAC protocol for underwater acoustic sensor network," *J. Distrib. Sensor Netw.*, vol. 9, no. 6, 2013, Art. no. 385471.

[2] L. Guangzhong and L. Zhibin, "Depth-based multi-hop routing protocol for underwater sensor network," in *Proc. 2nd Int. Conf. Ind. Mechatron. Automat. (ICIMA)*, vol. 2, May 2010, pp. 268–270.

[3] G. Acar and A. E. Adams, "ACMENet: An underwater acoustic sensor network protocol for real-time environmental monitoring in coastal areas," *IEE Proc.-Radar, Sonar Navigat.*, vol. 153, no. 4, pp. 365–380, Aug. 2006.

[4] J.-H. Cui, J. Kong, M. Gerla, and S. Zhou, "The challenges of building mobile underwater wireless networks for aquatic applications," *IEEE Netw.*, vol. 20, no. 3, pp. 12–18, May/Jun. 2006.

[5] M. Stojanovic and J. Preisig, "Underwater acoustic communication channels: Propagation models and statistical characterization," *IEEE Commun. Mag.*, vol. 47, no. 1, pp. 84–89, Jan. 2009.

[6] L. Liu, S. Zhou, and J.-H. Cui, "Prospects and problems of wireless communication for underwater sensor networks," *Wireless Commun. Mobile Comput., Underwater Sensor Netw., Archit. Protocols*, vol. 8, no. 8, pp. 977–994, 2008.

[7] M. Xu, G. Liu, and J. Guan, "Towards a secure medium access control protocol for cluster-based underwater wireless sensor networks," *Int. J. Distrib. Sensor Netw.*, vol. 11, no. 5, 2015, Art. no. 325474.

[8] K. Chen, M. Ma, E. Cheng, F. Yuan, and W. Su, "A survey on MAC protocols for underwater wireless sensor networks," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 3, pp. 1433–1447, 3rd Quart., 2014.

[9] F. Alfouzan, A. Shahrabi, S. M. Ghoreyshi, and T. Boutaleb, "Performance comparison of sender-based and receiver-based scheduling mac protocols for underwater sensor networks," in *Proc. 19th Int. Conf. Netw.-Based Inf. Syst. (NBiS)*, Sep. 2016, pp. 99–106.

[10] M. K. Park and V. Rodoplu, "UWAN-MAC: An energy-efficient MAC protocol for underwater acoustic wireless sensor networks," *IEEE J. Ocean. Eng.*, vol. 32, no. 3, pp. 710–720, Jul. 2007.

[11] N. Chirdchoo, W.-S. Soh, and K. C. Chua, "Aloha-based MAC protocols with collision avoidance for underwater acoustic networks," in *Proc. 26th IEEE Int. Conf. Comput. Commun. (INFOCOM)*, May 2007, pp. 2271–2275.

[12] F. Alfouzan, A. Shahrabi, S. M. Ghoreyshi, and T. Boutaleb, "An energy-conserving depth-based layering mac protocol for underwater sensor networks," in *Proc. IEEE 88th Veh. Technol. Conf. (VTC)*, Aug. 2018, pp. 1–6.

[13] C. L. Fullmer and J. Garcia-Luna-Aceves, "Floor acquisition multiple access (FAMA) for packet-radio networks," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 25, no. 4, pp. 262–273, 1995.

[14] P. Karn, "MACA—A new channel access method for packet radio," in *Proc. ARRL/CRRL Amateur Radio 9th Comput. Netw. Conf.*, Sep. 1990, pp. 134–140.

[15] M. Molins and M. Stojanovic, "Slotted FAMA: A MAC protocol for underwater acoustic networks," in *Proc. OCEANS-Asia Pacific*, May 2007, pp. 1–7.

[16] N. Chirdchoo, W.-S. Soh, and K. C. Chua, "MACA-MN: A MACA-based MAC protocol for underwater acoustic networks with packet train for multiple neighbors," in *Proc. IEEE Veh. Technol. Conf. (VTC Spring)*, May 2008, pp. 46–50.

[17] Y. Zhu, Z. Peng, J.-H. Cui, and H. Chen, "Toward practical MAC design for underwater acoustic networks," *IEEE Trans. Mobile Comput.*, vol. 14, no. 4, pp. 872–886, Apr. 2015.

[18] Y. Zhu, S. N. Le, Z. Peng, and J.-H. Cui, "DOS: Distributed on-demand scheduling for high performance MAC in underwater acoustic networks," Dept. Comput. Sci. Eng., Univ. Connecticut, Storrs, CT, USA, Tech. Rep. UbiNet-TR13-07, 2013.

[19] F. Alfouzan, A. Shahrabi, S. M. Ghoreyshi, and T. Boutaleb, "Graph colouring mac protocol for underwater sensor networks," in *Proc. IEEE 32nd Int. Conf. Adv. Inf. Netw. Appl. (AINA)*, May 2018, pp. 120–127.

[20] D. Pompili, T. Melodia, and I. F. Akyildiz, "A CDMA-based medium access control for underwater acoustic sensor networks," *IEEE Trans. Wireless Commun.*, vol. 8, no. 4, pp. 1899–1909, Apr. 2009.

[21] S. Jiang, "State-of-the-art medium access control (MAC) protocols for underwater acoustic networks: A survey based on a MAC reference model," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 1, pp. 96–131, 1st Quart., 2018.

[22] P. Xie and J.-H. Cui, "R-MAC: An energy-efficient MAC protocol for underwater sensor networks," in *Proc. Int. Conf. Wireless Algorithms, Syst. Appl. (WASA)*, Aug. 2007, pp. 187–198.

[23] E. M. Sozer, M. Stojanovic, and J. G. Proakis, "Underwater acoustic networks," *IEEE J. Ocean. Eng.*, vol. 25, no. 1, pp. 72–83, Jan. 2000.

[24] K. Kredo II, P. Djukic, and P. Mohapatra, "STUMP: Exploiting position diversity in the staggered TDMA underwater MAC protocol," in *Proc. IEEE INFOCOM*, Apr. 2009, pp. 2961–2965.

[25] I. F. Akyildiz, D. Pompili, and T. Melodia, "Challenges for efficient communication in underwater acoustic sensor networks," *ACM SIGBED Rev.*, vol. 1, no. 2, pp. 3–8, Jul. 2004.

[26] S. Climent, A. Sanchez, J. V. Capella, N. Meratnia, and J. J. Serrano, "Underwater acoustic wireless sensor networks: Advances and future trends in physical, MAC and routing layers," *Sensors*, vol. 14, no. 1, pp. 795–833, Jan. 2014.

[27] J. Cho, E. Shitiri, and H.-S. Cho, "Network allocation vector (NAV) optimization for underwater handshaking-based protocols," *Sensors*, vol. 17, no. 1, p. 32, 2016.

[28] X. Guo, M. R. Frater, and M. J. Ryan, "A propagation-delay-tolerant collision avoidance protocol for underwater acoustic sensor networks," in *Proc. OCEANS-Asia Pacific*, May 2007, pp. 1–6.

[29] B. Peleato and M. Stojanovic, "Distance aware collision avoidance protocol for ad-hoc underwater acoustic sensor networks," *IEEE Commun. Lett.*, vol. 11, no. 12, pp. 1025–1027, Dec. 2007.

[30] Y. Noh, P. Wang, U. Lee, D. Torres, and M. Gerla, "DOTS: A propagation delay-aware opportunistic MAC protocol for underwater sensor networks," in *Proc. 18th IEEE Int. Conf. Netw. Protocols (ICNP)*, Oct. 2010, pp. 183–192.

[31] A. A. Syed, W. Ye, J. Heidemann, and B. Krishnamachari, "Understanding spatio-temporal uncertainty in medium access with ALOHA protocols," in *Proc. 2nd Workshop Underwater Netw.*, 2007, pp. 41–48.

[32] L. Hong, F. Hong, Z. Guo, and Z. Li, "ECS: Efficient communication scheduling for underwater sensor networks," *Sensors*, vol. 11, no. 3, pp. 2920–2938, 2011.

[33] F. Alfouzan, A. Shahrabi, S. M. Ghoreyshi, and T. Boutaleb, "Efficient depth-based scheduling MAC protoco for underwater sensor networks," in *Proc. 9th Int. Conf. Ubiquitous Future Netw. (ICUFN)*, Jul. 2017, pp. 827–832.

[34] F. Alfouzan, A. Shahrabi, S. Ghoreyshi, and T. Boutaleb, "An efficient scalable scheduling MAC protocol for underwater sensor networks," *Sensors*, vol. 18, no. 9, p. 2806, 2018.

[35] J. Ma and W. Lou, "Interference-aware spatio-temporal link scheduling for long delay underwater sensor networks," in *Proc. 8th Annu. IEEE Commun. Soc. Conf. Sensor, Mesh Ad Hoc Commun. Netw. (SECON)*, Jun. 2011, pp. 431–439.

[36] C.-C. Hsu, K.-F. Lai, C.-F. Chou, and K. C.-J. Lin, "ST-MAC: Spatial-temporal MAC scheduling for underwater sensor networks," in *Proc. IEEE INFOCOM*, Apr. 2009, pp. 1827–1835.

[37] J. Yackoski and C.-C. Shen, "UW-FLASHR: Achieving high channel utilization in a time-based acoustic MAC protocol," in *Proc. 3rd ACM Int. Workshop Underwater Netw.*, 2008, pp. 59–66.

[38] K. Kredo, II, and P. Mohapatra, "Distributed scheduling and routing in underwater wireless networks," in *Proc. IEEE Global Telecommun. Conf. (GLOBECOM)*, Dec. 2010, pp. 1–6.

[39] F. Salva-Garau and M. Stojanovic, "Multi-cluster protocol for ad hoc mobile underwater acoustic networks," in *Proc. OCEANS*, vol. 1, Sep. 2003, pp. 91–98.

[40] K. B. Kredo, II, and P. Mohapatra, "A hybrid medium access control protocol for underwater wireless networks," in *Proc. 2nd Workshop Underwater Netw.*, 2007, pp. 33–40.

[41] P. Wang, C. Li, and J. Zheng, "Distributed minimum-cost clustering protocol for underwater sensor networks (UWSNs)," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2007, pp. 3510–3515.

[42] W.-H. Liao and C.-C. Huang, "SF-MAC: A spatially fair MAC protocol for underwater acoustic sensor networks," *IEEE Sensors J.*, vol. 12, no. 6, pp. 1686–1694, Jun. 2012.

[43] H. Chen, G. Fan, L. Xie, and J.-H. Cui, "A hybrid path-oriented code assignment CDMA-based MAC protocol for underwater acoustic sensor networks," *Sensors*, vol. 13, no. 11, pp. 15006–15025, 2013.

[44] I. F. Akyildiz, D. Pompili, and T. Melodia, "Underwater acoustic sensor networks: Research challenges," *Ad Hoc Netw.*, vol. 3, no. 3, pp. 257–279, Mar. 2005.

[45] J. Partan, J. Kurose, and B. N. Levine, "A survey of practical issues in underwater networks," *ACM SIGMOBILE Mobile Comput. Commun. Rev.*, vol. 11, no. 4, pp. 23–33, 2007.

[46] G. Fan, H. Chen, L. Xie, and K. Wang, "A hybrid reservation-based MAC protocol for underwater acoustic sensor networks," *Ad Hoc Netw.*, vol. 11, no. 3, pp. 1178–1192, May 2013.
[47] D. N. M. Dang, V. Nguyen, H. T. Le, C. S. Hong, and J. Choe, "An efficient multi-channel MAC protocol for wireless ad hoc networks," *Ad Hoc Netw.*, vol. 44, pp. 46–57, Jul. 2016.
[48] Y. Su, Y. Zhu, H. Mo, J.-H. Cui, and Z. Jin, "A joint power control and rate adaptation MAC protocol for underwater sensor networks," *Ad Hoc Netw.*, vol. 26, pp. 36–49, Mar. 2015.
[49] S. Alam and Z. J. Haas, "Coverage and connectivity in three-dimensional networks," in *Proc. 12th Annu. Int. Conf. Mobile Comput. Netw.*, 2006, pp. 346–357.
[50] A. Ghosh and S. K. Das, "Coverage and connectivity issues in wireless sensor networks: A survey," *Pervasive Mobile Comput.*, vol. 4, no. 3, pp. 303–334, 2008.
[51] P. Xie *et al.*, "Aqua-Sim: An NS-2 based simulator for underwater sensor networks," in *Proc. OCEANS MTS/IEEE Biloxi-Mar. Technol. Future, Global Local Challenges*, Oct. 2009, pp. 1–7.
[52] A. A. Syed, W. Ye, and J. Heidemann, "T-Lohi: A new class of MAC protocols for underwater acoustic sensor networks," in *Proc. IEEE 27th Conf. Comput. Commun. (INFOCOM)*, Apr. 2008, pp. 231–235.
[53] R. K. Jain, D. M. W. Chiu, and W. R. Hawe, "A quantitative measure of fairness and discrimination," Eastern Res. Lab., Digit. Equip. Corp., Hudson, MA, USA, 1984, pp. 2–7.

**FAISAL ABDULAZIZ ALFOUZAN** received the B.Sc. degree in computer science and software engineering from the University of Hail, Hail, Saudi Arabia, in 2007, and the M.Sc. degree in network security from Glasgow Caledonian University, Glasgow, U.K., in 2014, where he is currently pursuing the Ph.D. degree. His current research interests include underwater sensor networks, designing the medium access control (MAC) protocols, and distributed systems. He is a member of the IEEE Computer Society and VTS.

**ALIREZA SHAHRABI** received the B.Sc. and M.Sc. degrees in computer engineering from the Sharif University of Technology, Tehran, Iran, in 1991 and 1994, respectively, and the Ph.D. degree in computing science from the University of Glasgow, Glasgow, U.K., in 2003. Since 2001, he has been with the School of Engineering and Built Environment, Glasgow Caledonian University, where he is currently a Reader. He has published extensively in leading journals and well-established conferences. His current research interests include network protocols, wireless, mobile and sensor networks, and performance modeling and evaluation of parallel and distributed systems. He is a member of the IEEE Computer Society. He has been on the Editorial Board of some international journals and also served on the organizing and program committees of many international conferences and workshops.

**SEYED MOHAMMAD GHOREYSHI** received the B.E. degree from the Mazandaran University of Science and Technology, in 2009, the M.E. degree from the University of Tehran, in 2013, and the Ph.D. degree in computer networks and security from Glasgow Caledonian University, Glasgow, U.K. His current research interests include underwater sensor networks, designing the routing protocols, and void-handling techniques. He is a member of the IEEE Computer Society. He has served as a Reviewer for more than ten academic journals.

**TULEEN BOUTALEB** received the B.Eng. degree in electronics engineering and the Ph.D. degree in cellular mobile networks for telemetry from Glasgow Caledonian University, U.K., in 1995 and 2006, respectively. From 1995 to 1998, she was a Researcher on a European project working on the communications infrastructure. From 1998 to 2001, she was a Research Assistant working on a European project. Since 2001, she has been a Telecommunications Engineering Lecturer with the School of Engineering and Built Environment, Glasgow Caledonian University. Her research interests include cellular communication networks performance enhancement, messaging in VANET, WSN, and satellite communications for remote monitoring.